GRAPHICS

Dev C++ da graphics.h dosyasının yüklenmesi için <https://www.cs.colorado.edu/~main/bgi/dev-c++/> sayfasındaki yönergeleri takip edebilirsiniz. Ayrıca grafik işlemlerinin yapılacağı proje için “proje ayarlar” bölümdeki “parametreler” sekmesinde linker bölümüne aşağıdaki parametreleri eklemelisiniz.

-lbgi

-lgdi32

-lcomdlg32

-luuid

-loleaut32

-lole32

#include <cstdlib>

#include <iostream>

#include <graphics.h>

#include <math.h>

using namespace std;

int main()

{

int pol[100];

int x,y,x1=100,y1=100,x2=300,y2=300,i;

float t,n;

int poly[12]={350,450,350,410,430,400,350,350,300,430,350,450};

initwindow(600,600,"First Sample");

int EN = getmaxx(); // Maksimum X ekseni değri

int BOY = getmaxy(); //Maksimum Y ekseni değeri

float Pi = M\_PI;

int DaireCapi = 50;

char \*Metin;

int MetinBoyu;

int MetinYuksekligi;

/////////////////////Line///////////////////////////////////////////////////

int Merkez[2]={(int)round(EN/2),(int)round(BOY/2)};

line(0,0,EN,BOY); //line(x1,y1,x2,y2) komutu (x1,y1) noktasından (x2,y2) noktasına çizği çizer

line(0,BOY,EN,0);

////////////////////////////////////////////////////////////////////////////

///////////////////////Daire////////////////////////////////////////////////

Metin = "Daire";

MetinBoyu =(int)round(textwidth(Metin)/2);//textwidth metnin gelişğini vermektedir.

MetinYuksekligi =(int)round(textheight(Metin)/2); //textheight metnin yüksekliğini vermektedir.

circle(Merkez[0],Merkez[1],DaireCapi); // circle(x,y,r) merkezi (x,y) olan ve yarıçapı r olan çember çizer

outtextxy(Merkez[0]-MetinBoyu,Merkez[1]-DaireCapi-20,Metin); // outtextxy(x,y,str) (x,y) koordinatina str metinini yazar

////////////////////////////////////////////////////////////////////////////

///////////////////////Diktörtgen///////////////////////////////////////////

Metin = "Diktörtgen";

MetinBoyu =(int)round(textwidth(Metin)/2);

MetinYuksekligi =(int)round(textheight(Metin)/2);

rectangle(Merkez[0]-200,Merkez[1]-200,Merkez[0]+100,Merkez[1]-100);//sol üst ve sağ alt köşe koordinatları verilen diktörtgen çizer

outtextxy(Merkez[0]-100-MetinBoyu,Merkez[1]-220,Metin);

////////////////////////////////////////////////////////////////////////////

///////////////////////Elips////////////////////////////////////////////////

Metin = "Elips";

MetinBoyu =(int)round(textwidth(Metin)/2);

MetinYuksekligi =(int)round(textheight(Metin)/2);

ellipse(Merkez[0],Merkez[1],0,270,100,50);//ellipse(x,y,start,stop,br,sr) merkezi (x,y) koordinatinda olan sırasıyla başlangıç ve bitiş açısı ve büyük yarıçap ve küçük yarıcap verien elips çizer

outtextxy(Merkez[0]-MetinBoyu,Merkez[1]-225,Metin);

////////////////////////////////////////////////////////////////////////////

///////////////////////Elips////////////////////////////////////////////////

Metin = "Sektör";

MetinBoyu =(int)round(textwidth(Metin)/2);

MetinYuksekligi =(int)round(textheight(Metin)/2);

sector(150,400,30,300,100,50);//içi dolu elips parçası çizmektedir.

outtextxy(120,460,Metin);

////////////////////////////////////////////////////////////////////////////

int renk=COLOR(255,255,0);//COLOR fonksiyonu RGB değerlerini wibgi için düzenlemektedir.

setcolor(renk); // çizim rengini belirlemek için kullanılır

setbkcolor(3); // arka plan rengini belirlemektedir.

x = getx(); // aktif x konumu

y = gety(); // aktif y konumu

moveto(Merkez[0]-50,Merkez[1]-50); // belirlenen koordinatlara konumlanmak için kullanılıyor

lineto(Merkez[0]+50,Merkez[1]+50); //lineto(x,y) komutu mevcut cursor noktasından (x,y) noktasına çizgi çizer

lineto(Merkez[0]+100,Merkez[1]+100);

lineto(Merkez[0]+100,Merkez[1]-100);

renk=COLOR(0,25,255);//COLOR fonksiyonu RGB değerlerini wibgi için düzenlemektedir.

setcolor(renk); // çizim rebgini belirlemek için kullanılır

setbkcolor(2); // arka plan rengini belirlemektedir.

for (t=0;t<=1;t=t+0.001) // parametrik doğrunun denklemi ile doğru çizme

{

x=x1+(x2-x1)\*t;

y=y1+(y2-y1)\*t;

putpixel(x,y,YELLOW); // (x,y) koordinatina renk kodu verilmiş bir pixel koyar.

}

n=0.0001;

for (t=0;t<=6.28;t=t+n) // parametrik çember denklemi ile çizim;

{

x=300+200\*cos(t);

y=300+200\*sin(t);

putpixel(x,y,RED);

}

for (t=0;t<=6.28;t=t+n) // parametrik elipsin denklemi ile çizim;

{

x=300+100\*cos(t);

y=300+200\*sin(t);

putpixel(x,y,WHITE);

}

n=1/36;

for (i=0,x1=0;x1<=360;x1=x1+30,i++) // polgon oluşturma

{

pol[i]=(int)(300+150\*cos(6.28\*x1/360));

i++;

pol[i]=(int)(300+150\*sin(6.28\*x1/360));

}

pol[i]=pol[0];

i++;

pol[i]=pol[1];

drawpoly(6,poly);//verilen dizi deki noktalara ait çokgen oluşturmaktadır.

drawpoly(13,poly);

//////////////// klavyeden bir tuş alana kadar beklemek için////////////////

while (!kbhit( ))

{

delay(200);

}

////////////////////////////////////////////////////////////////////////////

return 0;

}

**FLOODFILL**

**Örnek:**

#include <cstdlib>

#include <iostream>

#include <graphics.h>

#include <math.h>

int main(int argc, char \*argv[])

{

initwindow(600,600,"First Sample");

setcolor(RED);

circle(300,300,100);

setfillstyle(SOLID\_FILL, BLUE);

floodfill(250,250,RED);

getch();

return 0;

}

**Örnek:**

int main(int argc, char \*argv[])

{

int maxx, maxy;

initwindow(600,600,"First Sample");

maxx = getmaxx();

maxy = getmaxy();

/\* select drawing color \*/

setcolor(getmaxcolor());

/\* select fill color \*/

setfillstyle(SOLID\_FILL, getmaxcolor());

/\* draw a border around the screen \*/

rectangle(0, 0, maxx, maxy);

/\* draw some circles \*/

circle(maxx / 3, maxy /2, 50);

circle(maxx / 2, 20, 100);

circle(maxx-20, maxy-50, 75);

circle(20, maxy-20, 25);

/\* wait for a key \*/

getch();

/\* fill in bounded region \*/

floodfill(2, 2, getmaxcolor());

getch();

return 0;

}

Temel Grafik Fonksiyonları

* **arc(x , y , stangele , endangle , radius)**
* **bar (left , top , right , bottom)**
* **bar3d (left , top , right , bottom, depth, topflag)**
* **circle (x , y , radius)**
* **drawpoly (n\_point , points)**
* **ellipse (x , y , stangele , endangle , xradius , yradius)**
* **fillellipse (x , y , stangele , endangle , xradius , yradius)**
* **fillpoly (n\_point , points)**
* **floodfill (x , y, border)**
* **line (x1 , y1 , x2 , y2)**
* **lineto (x , y)**
* **pieslice (x , y , stangele , endangle , radius)**
* **putpixel (x , y , color)**
* **rectangle (left , top , right , bottom, depth, topflag)**
* **sector (x , y , stangele , endangle , xradius , yradius)**
* **getbkcolor()**
* **getfillpattern(pattern)**
* **getfillsettings (fillinfo)**
* **getlinesettings(lineinfo)**
* **getmaxx()**
* **getmaxy()**
* **getpixel(x , y)**
* **getx()**
* **gety()**
* **moveto(x , y)**
* **setbkcolor(coor)**
* **setcolor(color)**
* **setfillpattern(pattern , color)**
* **setfillstyle(pattren , color)**
* **initwindow (width , height , title)**
* **kbhit()**
* **getimage(left , top , right , bottom,bitmap)**
* **putimage(left , top ,bitmap,op)**
* **outtext (text)**
* **outtextxy (x , y , text)**